Chapter 9: Product Installation: Special Cases

This chapter provides product installation information about specific cases. It discusses:

- how to install products requiring special privileges
- how to install into a local products area using the installation of **UPD** in AFS space
- how to install products into the AFS-space **UPS** products area

9.1 Installing Products that Require Special Privileges

Certain products supplied by the Computing Division need "special configuration" which can only be performed by a suitably privileged account. This is described in TN0092 *What does 'InstallAsRoot' Really Mean?*. The text here is adapted from that document.

For these particular products, listed below, at some point during the installation process you will be prompted to login as *root* and run the command[1]:

```
% ups installasroot <product> <version> [<options>]
```

This command would then proceed to run the privileged installation actions. The `INSTALL_NOTE` file should provide instructions for you if this is necessary. See TN0092 *What does 'InstallAsRoot' Really Mean?* for information on this topic.

Examples of products requiring configuration by a privileged account include:

python, **perl** require that files and symlinks be created in `/usr/local/bin` for the convenience of users and system administrators (so that **perl** and **python** are always accessible, even if not previously setup).

1. By convention, the products' table files generally contain an installAsRoot action, which gets executed via this command. For particularly complicated products, the installAsRoot action my point to other developer-defined actions, and you may be instructed to run one or more customized commands instead of **`ups installasroot`**.

tcsh, **bash**	require that files be copied to `/usr/local/bin` with proper permissions and ownership (for security reasons)
ssh	requires that configuration files and binaries be copied to system areas
kerberos	requires that configuration files and binaries be copied to system areas; also requires **suid** on certain files under the product area $KERBEROS_DIR itself.
systools	requires that **suid** permissions be set on various **cmd** plug-in scripts under the $PRODUCTS area.

On many systems, `/usr/local` and/or the $PRODUCTS area are NFS-mounted. For security reasons, these areas may not in fact be writable by the *root* account on the node where the product installation is taking place.

Note that in AFS file systems, *root* access is usually insufficient to guarantee write access. At present, however, there are no products known to require an admin token for their installAsRoot actions.

If you are instructed to issue a special installation command, e.g.,:

```
% ups installAsRoot <product> <version>
```

assume that you need full write access to the following locations:

`/usr/local`	Scripting languages, local utilities, and certain security tools will require symlinks and/or files under `/usr/local/bin` (or `/usr/local/etc`). Bear in mind that in a mixed-platform cluster, `/usr/local` will typically comprise a set of directories, one for each type of system.
$PRODUCTS	More accurately, *root* may need to write/modify configuration and/or log files under the area where products are installed. This is determined by the system's **UPD** configuration, usually found in the file `$PRODUCTS/.updfiles/updconfig`.
local system disk	Security tools, system administration tools, web servers, and so on, may need to write configuration files into system areas such as `/etc` and `/var`.

If access to other areas is required, it should be noted in the product's `INSTALL_NOTE` file. The steps to take in order to ensure that areas listed above are writable will vary depending on the particular configuration of each system, and are left to the system administrator.

For some older versions of products, a symbolic link gets created in `/usr/local/products` whenever a new instance is declared to the database. These products will need to be configured on each machine with a unique `/usr/local` area. This packaging philosophy has been phased out.

9.2 Installing Locally Using UPD from AFS-Space

Systems running AFS can be configured to provide access both to locally installed/declared products and to products in AFS space without maintaining **UPS/UPD** in the local database. This configuration is described in section 13.2 *Configuring a Local Database to Work With AFS*.

The local database is usually given a standard name in common use at Fermilab:

`/fnal/ups/db`	standard for several product server bootstrap configurations
`/local/ups/`	standard for Fermi RedHat Linux
`/usr/products/`	another popular naming convention
`/usr/products/CMSUN1/`	naming convention for CMS local databases

The database must point to a local **UPD** configuration file with appropriate product location definitions.

With no locally installed **UPS/UPD**, you'll need to invoke the AFS installation of **UPD** to install a product into the local products area. If the first or only database listed in $PRODUCTS that contains **UPD** is the AFS database, then you need no database specifier in the **`setup`** command. Run it normally:

```
% setup upd
```

Assuming that $PRODUCTS lists your local database first, you can run **`upd install`** without any database option and your product will go into the local database (assuming the **UPD** configuration is set accordingly). If $PRODUCTS doesn't list yours first, include the **`-z`** option in the **`upd install`** command, e.g.,:

```
% upd install -z /path/to/local/db <product> <version> ...
```

In terms of where they get installed and declared, dependencies are treated the same as the main product. If you want to install and declare only the main product locally (e.g., for development) but you want to keep all of its dependencies in AFS space, use the syntax:

```
% upd install -j [-z /path/to/local/db] <product> <version> ...
```

to install only the main product. Then, as needed, install the dependencies in AFS-space; see section 9.3 *Installing Products into AFS Space*.

9.3 Installing Products into AFS Space

9.3.1 Overview

A single AFS *volume* is intended to contain instances for all flavors of a particular **UPS** product-version combination. For each product, there is a read/write volume into which the product must be installed:

`/afs/.fnal.gov/ups/<product>/<version>`

(note the dot preceding `fnal`). There is a process called *releasing a volume* which replicates the read/write product volume into *read-only clones.* Replication avoids any single point of failure for a product and provides more robust service. Multiple frozen read-only copies of the product areas are kept on several AFS server machines. We want users running **`setup`** to access these redundant, read-only volumes of products. Otherwise, the benefit of cloning is wasted. A product must therefore be declared to **UPS** via its read-only pathname (notice the absence of a dot preceding `fnal`):

`/afs/fnal.gov/ups/<product>/<version>`

The AFS-space `updconfig` file is configured to unwind products into the read/write area (via the UNWIND_PROD_DIR definition), and then release them to the read-only volumes, using the `upd_volrelease` script (via the PREDECLARE action). As the action name suggests, this happens before declaring the product. When **`ups declare`** gets called, the PROD_DIR_PREFIX in the AFS-space `dbconfig` file ensures that the read-only pathname gets declared.

Installations into AFS space should be made from one of the interactive nodes of the *fnalu* cluster, preferably a SunOS node.[1] The *fnalu* nodes have the **arcd** daemon running and supply the `upd_volrelease` script, both of which are required for AFS installations. *fsui02.fnal.gov* is the machine on which it works most consistently. If you need to use a non-SunOS node, use **`upd install -H <target_flavor>`** to set the flavor. The userid *products* and the groups *uas* and *upsdatabase* are allowed to install into AFS space. Other users/groups must create a helpdesk ticket requesting permission to release a volume.

1. On some AFS nodes the `upd_volrelease` script is missing from `/usr/local/bin`, and on others the setup work for it has not been completed.

9.3.2 Request a Product Volume

Only AFS administrators are allowed to create product volumes. To request a product volume, contact Customer Support at *helpdesk@fnal.gov.* Customer Support will forward the message appropriately. Recall that all the instances for the different flavors of a particular product-version pair go into a single AFS volume. Your request needs to include:

- the product name
- the product version
- the combined size of the various instances that will go into the volume
- the AFS user and group(s) who need write access

An AFS administrator will create a volume writable by you (the requestor) and your group, and notify you when it's ready. The product instances can be installed in the volume as soon as it is created.

9.3.3 Install your Product

Be sure you have a Kerberos ticket and an AFS token first! (See the Strong Authentication documentation at `http://www.fnal.gov/docs/strongauth/index.html`.) Install each instance of your product (all of the same version) using the **`upd install`** command (documented in Chapter 6: *Installing Products Using UPD*). Specify the read-only path for the database as shown here:

```
% upd install -z /afs/fnal.gov/ups/db <product> <version> \
  -f <flavor> [-q <qualifierList>] ...
```

9.3.4 Post-Installation Steps

Configure/Tailor the Product

Because the product areas as declared in AFS space are read-only, if your product requires configuration or tailoring, you must execute these commands using the read/write path name, e.g.,:

```
% ups configure -r /afs/.fnal.gov/... <product> <version> \
  -f <flavor>...
```

```
% ups tailor -r /afs/.fnal.gov/... <product> <version> \
  -f <flavor> ...
```

For some products, notably **perl** and **python**, the configure script/action checks for `/afs/` and makes the appropriate path change, e.g.,:

```
DEST_DIR=‘echo $UPS_PROD_DIR | sed -e
‘s;/afs/fnal.gov;/afs/.fnal.gov;’`
```

Create Symbolic Links

If the product needs the ability to write into any areas under its product root directory during normal use, then you need to symbolically link these areas. If the area must be shared, link to the read/write area. If not, you can link to some area on non-AFS writable disk (e.g., under `/tmp` or `/var`).

For example, say the product **fred** in AFS space needs to write into `$FRED_DIR/log` which is read-only. Go into the read/write $FRED_DIR area, remove the log directory and create a link for the area in which to write (e.g., `/var/tmp/fred`) in the read-write area, e.g.,:

```
% ln -s /var/tmp/fred /afs/.fnal.gov/ups/fred/v1_0/SunOS/log
```

You will then need to release the volume, as described below. Your read-only replicas will contain the link.

If links are made to a non-AFS writable disk, check the SETUP action in the product’s table file; it should ensure that the specified area exists at product setup. E.g., if linking to `/var/tmp/fred`:

```
Action = setup
...
Execute( test -d /var/tmp/fred || (mkdir /var/tmp/fred;
         chmod 777 /var/tmp/fred), NO_UPS_ENV) (all on one line in
real file)
```

Rerun the Volume Release

If you have configured and/or tailored the product, or if you have added symbolic links, you need to manually rerun the `upd_volrelease` script to re-release the product volume, e.g.,:

```
$ upd_volrelease /afs/.fnal.gov/ups/<product>/<version>
```

unless the product's actions already take this into account (look for `upd_volrelease` calls in the table file's CONFIGURE action).

Note that it doesn't hurt to re-release a product volume several times in a row, so if you're not sure, just rerun it.

To save time, configure and/or tailor all the flavors of your product version first, and then run the **`upd_volrelease`** command once at the end.